Grassmannian Representation of Motion Depth for 3D Human Gesture and Action Recognition

Rim Slama, Hazem Wannous, Mohamed Daoudi

Grassmannian Representation of Motion Depth for 3D Human Gesture and Action Recognition

Rim Slama
LIFL (UMR CNRS 8022), France
University of Lille 1, France
Email: rim.slama@telecom-lille.fr

Hazem Wannous
LIFL (UMR CNRS 8022), France
University of Lille 1, France
Email: hazem.wannous@telecom-lille.fr

Mohamed daoudi
LIFL (UMR CNRS 8022), France
Institut Mines-Télécom / Télécom Lille, France
Email: mohamed.daoudi@telecom-lille.fr

***Abstract*—Recently developed commodity depth sensors open up new possibilities of dealing with rich descriptors, which capture geometrical features of the observed scene. Here, we propose an original approach to represent geometrical features extracted from depth motion space, which capture both geometric appearance and dynamic of human body simultaneously. In this approach, sequence features are modeled temporally as subspaces lying on Grassmannian manifold. Classification task is carried out via computation of probability density functions on tangent space of each class tacking benefit from the geometric structure of the Grassmaniann manifold. The experimental evaluation is performed on three existing datasets containing various challenges, including MSR-action 3D, UT-kinect and MSR-Gesture3D. Results reveal that our approach outperforms the state-of-the-art methods, with accuracy of 98.21% on MSR-Gesture3D and 95.25% on UT-kinect, and achieves a competitive performance of 86.21% on MSR-action 3D.**

I. Introduction

Thanks to the advancement in information technologies, effective and inexpensive depth video cameras, like Microsoft Kinect or Asus Xtion PRO LIVE, are increasingly used in the domain of computer vision. One of the most active research area in this domain is human action recognition. The motivation behinds the great interest for action recognition is the large number of possible applications in: consumer interactive entertainment and gaming [1], surveillance systems [2], life-care systems and smart home systems [3].

The main challenges in almost action recognition system are the accuracy of acquisition data and the dynamic modeling of the movements. The major problems in video based human action recognition which can alter the way actions are perceived, and consequently be recognized, are: occlusions, shadows and background extraction, lighting condition variations and viewpoint changes. The recent release of consumer depth cameras, like Microsoft Kinect, has significantly alleviate these difficulties that reduce the action recognition performance in 2D video. These cameras provide, in addition to the RGB image, depth stream allowing to discern changes in depth. In this paper, we address the problem of modeling and analyzing human motion in depth sequences. Particularly, we recast action recognition problem as a statistical problem on Grassmannian manifold.

In recent years, many approaches dealing with human action and gesture recognition in depth sequence have received growing attention. These approaches can be categorized into 3D joint-based approaches, depth-based approaches and hydride approaches.

First methods used for activity recognition from depth sequences have tendency to extrapolate techniques already developed for 2D video sequences to depth ones. These approaches use all points in depth map sequences as a gray pixels in images to extract meaningful spatiotemporal descriptors. In [4], depth maps of a sequence are projected onto the three orthogonal Cartesian planes and the contours of the projections are sampled for each frame as *bag-of-points* to model the dynamics of the actions as an *action graph*. Vieira et al. [5], represent each depth map sequence as a 4D grid by dividing the space and time axes into multiple segments in order to extract Spatio-Temporal Occupancy Pattern features (STOP). Also in Wang et al. [6], the action sequence is considered as a 4D shape and Random Occupancy Pattern (ROP) features are extracted. In [7], the average difference between the depth frames is computed and summarized in a single Depth Motion Maps (DMM), from which a a Histograms of Oriented Gradients features (HOG) are extracted. Each DMM is generated by projecting depth map from the sequence onto three orthogonal Cartesian planes, computing a motion energy by thresholding the difference between two consecutive maps, and stacking the energies for each projection. A classical SVM classifier is finally used for recognition. By the same token, Oreifej et al. [8] compute a 4D histogram over depth, time, and spatial coordinates capturing the distribution of the surface normal orientation. This histogram is created using 4D projectors allowing quantification of 4D space.

The availability of 3D sensors has recently made possible to estimate 3D positions of body joints using depth sensors through a real-time 3D joint position prediction system [9]. Joint-based approaches have then become popular and many approaches in the literature propose to model the dynamic of the action using these features. Xia et al. [10] comput histograms of the locations of 3D joints as a compact representation. Yang et al. [11] proposes to use eigenjoints which combine action information including static posture, motion, and offset.

Some hybrid approaches combining both 3D joint data features and depth information were recently introduced. They are trying to take benefit from positive aspects of both approaches. Azary et al. [12] propose spatiotemporal

descriptors as time-invariant action surfaces, combining features extracted using radial distance measures and 3D joint tracking. In [13], local features are computed from patches around the body joints. A structure of a particular conjunction of the features is then defined for a subset of the joints as an *actionlet*. Finally, an action is represented as a linear combination of the *actionlets*, where the discriminative weights are learnt via a multiple kernel learning method. In Oreifej et al. [8], a spatiotemporal histogram (HON4D) is computed over depth sequences to encode the distribution of some 4D normals. Similarly to [13], HON4D histograms are computed around joints and concatenated in a spatiotemporal descriptor of the sequence to provide the input of an SVM classifier.

In recent years, variety of techniques reformulating computer vision problems over non-Euclidean spaces, such as Riemannian manifolds, have received growing attention. Such state-of-the-art manifold techniques is presented by Turaga et al. [14], [15].

In this paper, data representation considers the geometry of space and incorporates the intrinsic nature of the data. In a such framework, which is 3D depth-based, both geometric appearance and dynamic of human body are captured simultaneously. The main contributions of this paper are: First, a novel proposed approach for gesture and activity recognition from depth sequences, in which motion is represented as a local displacement of the normal vector orientation lying on Grassmaniann manifold. Second, a learning algorithm is introduced using the notion of the tangent space on the manifold, where the classification process is then performed as a function of probability density by Truncated Warped Gaussian on specific-class tangent spaces. Finally, it is demonstrated how this representation capture both geometric appearance and dynamic of human body simultaneously without any information about joint position.

The rest of the paper is organized as follows. In Section II, our approach is described and then Grassmannian manifold and learning algorithm are introduced. Section III presents the experimental results and inroduce the datasets used for evaluations. Section IV concludes the paper.

II. Approach

Our approach describes a motion of depth images as sequence features modeled temporally as subspaces lying on Grassmannian manifold. First, we compute local oriented displacements of each sequence and represent it as a time series of angle orientations. Second, action sequence represented by its time series is modeled as an autoregressive and moving average model (ARMA). An observability matrix is then computed using ARMA parameters and represented by its orthonormal basis using Gram shmith. This last operation allows representing the action as a subspaces lying on Grassman manifold. Finally, each action on this manifold is learnt by its mean and covariance according to its class-specific tangent space.

A. 3D oriented displacement features

With a depth sensor, the distance between the pixel position and the depth sensor z, is obtained and quantized into 11-bit digits. The depth information captured by a depth sensor is usually called the depth image. We denote each pixel in the depth image as $P = (x; y; z)$. Let $I = [I(1), I(2), ..., I(t), I(\tau)]$ denotes the depth sequence. This sequence can be seen as a 4D surface S in the 4D space if we considere a function $\mathbb{R}^3 \longrightarrow \mathbb{R}^1 : z = I(x; y; t)$ [8].

Since the orientation of the normal vector, at every surface point, can describe the surface of an object, the local 4D geometry characteristics (Depth + motion) can be represented as a local displacement of the normal vector orientation. The normals of this surface are given by a derivation of $S(x, y, z, t)$ where $S(x, y, z, t) = f(x, y, t) - z = 0$. Thus, $n = \bigtriangledown S = (\frac{\partial z}{\partial x}; \frac{\partial z}{\partial y}; \frac{\partial z}{\partial t}; -1)^T = (n_x, n_y, n_t, -1)$ if we follow the same demonstration of Tang et al. [16]. Experimentally $\frac{\partial z}{\partial x}$, $\frac{\partial z}{\partial y}$ and $\frac{\partial z}{\partial t}$ are calculated using the finite difference approximation respectively:

$$\begin{aligned} \frac{\partial z}{\partial x} &\simeq I(x - Diff, y, t) - I(x + Diff, y, t) \\ \frac{\partial z}{\partial y} &\simeq I(x, y - Diff, t) - I(x, y + Diff, t) \\ \frac{\partial z}{\partial t} &\simeq I(x, y, t) - I(x, y, t + 1) \end{aligned} \tag{1}$$

where $Diff$ is a positif value of displacement on image matrix. Encoding orientation information of this normal is more meaningful for describing the surface, than (x,y,z,t) coordinates. Thus, these local oriented displacements can be parametrized using spherical coordinates represented as 3 angles Θ, Φ and Ψ descibing respectively zenith angle, azimuth angle and inclination angle. These angles, which are illustrated in Figure 1, are computed as follows:

$$\begin{aligned} \Theta &= \tan^{-1}(\sqrt{n_x^2 + n_y^2 + n_t^2}) \\ \Phi &= \tan^{-1}(\frac{n_y}{n_x}) \\ \Psi &= \tan^{-1}(\frac{n_t}{\sqrt{(n_x^2 + n_y)}}) \end{aligned} \tag{2}$$

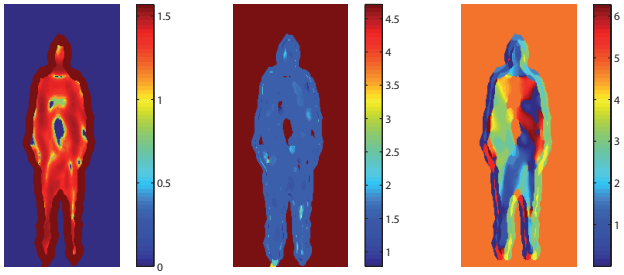

Fig. 1. 3D angles illustration. From the left to the right the angles Θ, Φ and Ψ.

B. Temporal modeling

After feature extraction step, the sequence of depth images can be represented as a time series model of features: $F = [f(1), f(2), ..., f(\tau)]$.

3D oriented displacement features computed on each image are linearized on a vector $f(t)$ for modeling the time series.

Let $\Psi(x, y, t)$ denotes the angle orientation of a pixel computed between $I(t)$ and $I(t + 1)$. $f(t) =$

$[\Psi(1,1,t), \Psi(1,2,t), \cdots \Psi(n,m,t)]$, with $n \times m = p$ the resolution of the image I. $F = f(1), f(2),, f(T)$, with T the number of frames -1 and $f \in \mathbb{R}^p$. A motion sequence can then be seen as a matrix representing a time-series from angle features. Dynamics and continuity of movement implies that action can not be resumed as a simply set of oriented 3D normal because of the temporal information contained in the sequence. Instead of directly using original time-series data, we believe that a linear dynamic system, like that often used for dynamic texture modeling, is essential before manifold analysis. Therefore, to capture both the spatial and the temporal dynamics of a motion, linear dynamical system characterized by ARMA models, will be applied to the time-series matrix M. The dynamic captured by the ARMA model during an action sequence M can be represented as:

$$\begin{aligned} p(t) &= Cz(t) + w(t), \quad w(t) \sim N(0, R), \\ z(t+1) &= Az(t) + v(t), \quad v(t) \sim N(0, Q) \end{aligned} \tag{3}$$

where $z \in \mathbb{R}^d$ is a hidden state vector, $A \in \mathbb{R}^{d \times d}$ is the transition matrix and $C \in \mathbb{R}^{3*J \times d}$ the measurement matrix. w and v are noise components modeled as normal with mean equal to zero and covariance matrix $R \in \mathbb{R}^{p \times p}$ and $Q \in \mathbb{R}^{d \times d}$ respectively. The goal is to learn parameters of the model (A, C) given by these equations. Let $U \sum V^T$ be the singular value decomposition of M. Then, the estimated model parameters A and C are given by: $\hat{C} = U$ and $\hat{A} = \sum V^T D_1 V (V^T D_2 V)^{-1} \sum^{-1}$, where $D_1 = [0 \quad 0, I_{\tau-1} \quad 0]$ and $D_2 = [I_{\tau-1} \quad 0, 0 \quad 0]$ where I represents the identity matrix. Comparing two ARMA models can be done by simply comparing their observability matrices. The expected observation sequence generated by an ARMA model (A,C) lies in the column space of the extended observability matrix given by $\bigcirc_\infty^T = [C^T, (CA)^T, (CA^2)^T, ...]^T$. This can be approximated by the finite observability matrix $\bigcirc_m^T = [C^T, (CA)^T, (CA^2)^T, ..., (CA^2)^m]^T$ [17]. The subspace spanned by columns of this finite observability matrix correspond to a point on a Grassmannian manifold $G_{n \times d}$.

C. Grassmann analysis

Manifold analysis has been widely used with success by various disciplines and for several applications including image set matching, face recognition and action recognition. In this work we are interested in Grassmannian manifolds. Two points U_1 and U_2 on $G_{n,d}$ are equivalent if one can be mapped into the other one by $d \times d$ orthogonal matrix [18]. In other words, U_1 and U_2 are equivalent if the d column of U_1 are rotations of U_2 The minimum length curve connecting these two points is the geodesic between them computed as :

$$d_{geod}(U_1, U_2) = \| \theta_1, \theta_2, \cdots, \theta_i, \cdots, \theta_d \|_2 \tag{4}$$

where θ_i is the principal angle vector which can be computed through the SVD of $U_1^T U_2$.

Grassman analysis provides a natural way to deal with the problem of sequence matching. Specially, as $G_{n,d}$ allows to represent a sequence by a point on its manifold and offer tools to compare and do statistics on this manifold. The classification problem of matching sets of motions represented by a collection of features can be transformed to point classification problem on $G_{n,d}$.

D. Learning on the Grassmann manifold

The Karcher Mean enables computation of a mean representative for a cluster of points on the Grassmann manifold. The algorithm exploits exp and log maps in a predictor/corrector loop until convergence to an expected point [19].

The karcher Mean enables computation of a mean representative for a cluster of points on the manifold. This mean should belong to the same space as the given points. In our case, we need karcher mean to compute averages on the Grassman manifold. Let μ denotes a mean obtained by the karcher mean on a set on sequences $\{U_i\}_{i=1:N}$ belonging to the same class of action. In addition to this mean, we look for the standard deviation value σ between all actions in each class of training data. The σ must be computed on $\{V_i\}_{i=1:N}$ where $V = exp_\mu^{-1}(U_i)$ are the projections of actions from the Grassmannian manifold into the tangent space defined on the mean μ. The key idea here is to use the fact that the tangent space $T_\mu(G_{n,d})$ is a vector space.

Thus, we can estimate the parameters of a probability density function such as a Gaussian and then use the exponential map to wrap these parameters back onto the manifold using exponential map operator (see Figure 2).

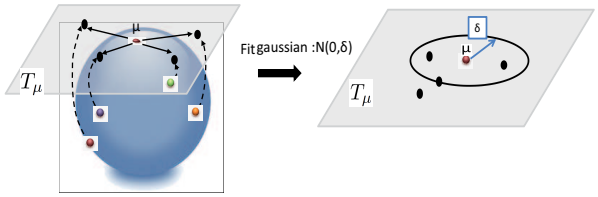

Fig. 2. Algorithms for estimating class-conditionals on class-specific poles.

However, the exponential map is not a bijection for the Grassmannian manifold. In fact, a line on tangent space with infinite length, can be warped around the manifold many times. Thus, some points of this line are going to have more than one image on $G_{n,d}$. It becomes a bijection only if the domain is restricted. Therefore, we can restrict the tangent space by a truncation beyond a radius of π in $T_\mu(G_{n,d})$. By truncation, the normalization constant changes for multivariate density in $T_\mu(G_{n,d})$. In fact, it gets scaled down depending on how much of the probability mass is left out of the truncation region.

Let $f(x)$ denote the probability density function (pdf) defined on $T_\mu(G_{n,d})$ by :

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-x^2}{2\sigma^2}} \tag{5}$$

After truncation, an approximation of f gives:

$$\hat{f}(x) = \frac{f(x) 1_{|x|<\pi}}{z} \tag{6}$$

where z is the normalization factor :

$$z = \int_{-\pi}^{\pi} f(x) 1_{|x|<\pi} dx \tag{7}$$

Using Monte Carlo estimation it can be demonstrated that the estimation of z is given by:

$$\hat{z} = \frac{1}{N}\sum_{i=1}^{N} 1_{|x_i|<\pi} \tag{8}$$

In practice, we employ wrapped Gaussians in each class-specific tangent space. Separate tangent space is considered for each class at its mean computed by Karcher Mean algorithm. Predicted class of an observation point is estimated in these individual tangent spaces. In the training step, the mean, standard deviation and normalization factor in each class of actions are computed. The predicted label of unknown class action is estimated as a function of probability density in class-specific tangent spaces (see Algorithm 1).

Algorithm 1: pdf classification by TWG on class-specific tangent space

****** *Training* ******

Input: N training actions as points on $G_{n,d}$, belonging to k classes: $D = \{U_i, l_j, \}_{i=1:N, j=1:k}$
Output: M
ultiplication factor ($\hat{z_j}$) and standard deviation σ_j for each class
for *j=1 : k* **do**
- **1-** Compute the karcher mean μ_j of the j^{th} class
- **2-** Compute the standard deviation σ_j of v_i
- **3-** Sample a large number of points from the Gaussian, $N(0, \sigma_j)$, estimated by the fitted Gaussian to the set of points $\{V_i\}_j^l$.
- **4-** Count N_π points from N generated ones that lie within a distance π from the origin of $T_{\mu_j}(G_{n,d})$
- **5-** Compute multiplication factor $\hat{z}_j = N_\pi / N$
- **6-** Adjust normalization factor for the j^{th} class conditional density $\hat{f}$

****** *Testing* ******

Input: U: unknown action, $\hat{z_j}$, σ_j
Output: l: class label
for *j=1 : k* **do**
- **1-** Compute $v_j = log_{\mu_j}(U)$
- **2-** Compute the probability of belonging to class j: $\hat{f}_j = f(v_j)/\hat{z}_i$

3- Predict the class label l of U which belong to the class with maximum probability $\hat{f}_j$.

III. Experimental results

This section summarizes our obtained results and provides an analysis of the performances of our proposed approach tested on several datasets and compared with state-of-the-art methods. The evaluation is made on three publicly datasets, containing various challenges, including: MSR-action 3D [4], UT-kinect [10] and MSR-Gesture3D [6]. Example frames from these datasets are shown in Figure 3.

A. MSR-Action 3D dataset

MSR-Action 3D [4] is a public dataset of 3D action captured by a depth camera. Sequences of this dataset consist of a set of temporally segmented actions where subjects are facing the camera and the background is pre-processed clearing discontinuities. Despite this, it is also a challenging dataset since many activities appear very similar due to small inter-class variation.

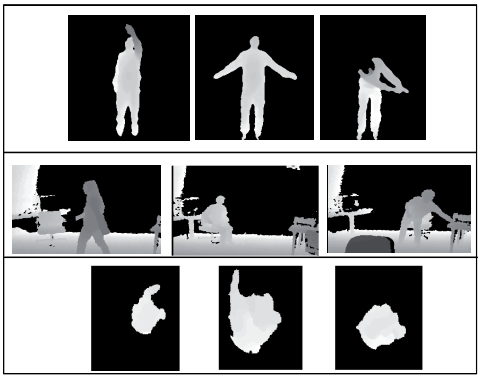

Fig. 3. Example frames from different actions obtained respectively, from top to bottom, from MSR-action 3D, UT-kinect and MSR-Gesture3D dataset.

Angle normals computation is performed on cropped area around models. For each frame normal angles features computed on cropped area gives 3800 features.

To reduce this feature dimension, we learnt a low dimension features using PCA. This dimension reduction allows working with features with lower size and also avoid the manipulation of long vectors, whose computation is costly, containing redundant information . The feature vector initially contains 3800 features. This feature dimension can be reduced to 500 while kiping 100% of informations. In our experiments we chose to reduce the feature vector to 200 by kipping 87% of the information.

This final feature vector is computed on each frame allowing to build the time series that characterize the action. Then, we fit an ARMA model and we compute observability matrix and its basis which represents action as a point on $G_{n\times d}$ with $n = 200 \times m$ and $d = m = 16$.

Table I summarized accuracies of the state-of-the-art methods. To evaluate our approach, we followed the same experimental setup as in Oreifej et al. [8] and Jiang et al. [13], where first five actors are used for training and the rest for testing.

Method	accuracy %
Histograms of 3D Joints [20]	78.97
Eigen Joints [11]	82.33
DMM-HOG [7]	85.52
HON4D [8]	85.80
Random Occupancy patterns [6]	86.50
HOH4D + D_{disc} [8]	88.89

TABLE I. Recognition accuracy (in %) for the MSR-Action 3D dataset obtained by the most kown state-of-the-art approaches.

We firstly choose to test the efficiency of normal angles seperately, then we use the 3 angles as feature for each image.

We note that our method using Ψ angles as features to model the time series gives the best recognition rate comparing to Θ, Φ or even the three angles together as illustrated in

II. As summarized in II,our approach achieves an accuracy of 86.21%, just below the best method from the state-of-the-art proposed by Oreifej et al. [8]. Knowing that our approach is based on only 3D oriented displacement features without any information about 3D joint poisitions, compared to other approaches, such as [8] and [6] which use the depth information around joint locations.

Method	accuracy %
θ angle	79.02
Φ angle	84.14
$\theta + \Psi + \Phi$ angles	85.19
Ψ angle	**86.21**

TABLE II. RECOGNITION ACCURACY (IN %) FOR THE MSR-ACTION DATASET USING DIFFERENT ORIENTATION DISPLACEMENT ANGLES.

All results in the rest of experiments are obtained using only Ψ angle as feature to represent the time series.

Figure 4 gives more details about recognition per class. The first observation is that using our approach about 10 actions are 100% correctly classified. The second observation is on the misclassified actions which are mainly 3 actions: 'Hammer' confused with 'draw X', 'hand catch' confused with 'draw tick' and 'hight serve' with 'hight throw'.

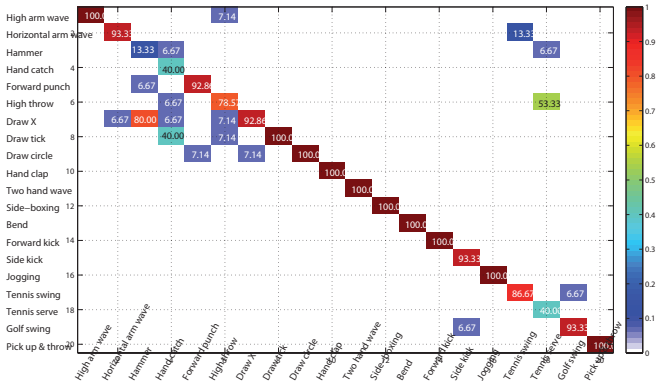

Fig. 4. Confusion matrix for the proposed approach on MSR-Action 3D dataset.

B. Ut-Kinect dataset

Ut-kinect dataset [10] contains 10 different actions, including: walk, sit down, stand up, pick up, carry, throw, push, pull, wave and clap hands.Each action is performed two times by ten subjects. Sequences are taken using one kinect in indoor settings and their length vary from 5 to 120 frames. We use this dataset because it contains several challenges like view change, significant variation in the realisation of the same action, variation in duration of actions and occlusions. The dataset contains both RGB and depth sequence images but for our experiments we use only depth sequences which resolution is 320×240.

To compare our results with state of the art approaches, we follow experiment protocol proposed in [10] and followed in [21]. The protocol is leave-one-out cross-validation. Table III compared the recognition accuracy produced using our approach and previous systems. As shown, our approach outperforms the tow methods. Indeed, all the actions are correctly classified with a score more than 90%. Some actions in this dataset include human-object interaction (pick-up, carry, throw), which Devanne et al. [21] fail to correctly classify these actions since their approach rely totally on skeleton features. Thus, actions like throw (action with object interation) and push (action without object iteration) are classified the same.

However, our approach, since it is based on features computed on depth images, overcome this problem. The flaw of Xia's method [10] is that complex actions effects adversely his HMM classification when the number of training samples is small.

Method	accuracy %
Histogram of 3D joints [10]	90.92
Space-time Pose Representation [21]	91.5
Our approach	**95.25**

TABLE III. RECOGNITION ACCURACY (IN %) FOR THE UT-KINECT DATASET USING OUR APPROACH COMPARED TO THE PREVIOUS APPROACHES.

C. Gesture 3D dataset

The gesture 3D dataset [22] contains 336 depth sequences of 12 hand gesture defined by American sign language (ASL). These gestures are: bathroom, blue, finish, green, hungry, milk, past, pig, store, where, j, z. Following experiment setup used by Kurakin et al. [22], the protocol used for evaluation is Leave-one-subject-out-cross-validation. We note that the resolution of depth maps is different from one sequence to an other. In order to ensure the consistensly of the scale, each depth sequence is resized to the same size given images with resolution 50×50. Accuracies obtained with our approach and using state of the art approaches are summarized in table IV. Our performance is better than HON4D presented by Oreifej et al. [8]. This can be explained by the facrt that HON4D computes histograms of 4D normals while we are using directly the normal information and he is segmenting the sequence into fixed number of cells which is very sensitive to change in exceution rate. Finally, using subspaces allows being robust to noise and missing data and in this dataset, several frames are either empty or with noise.

Method	accuracy
Oreifej et al. [8]	92.45
Jiang et al. [7]	88.50
Yang et al. [6]	89.20
Klaser et al. [23]	85.23
Our approach	**98.21**

TABLE IV. THE PERFORMANCE ON MSR HAND GESTURE 3D DATASET COMPARED TO PREVIOUS APPROACHES.

IV. CONCLUSION

This paper addressed the problem of human gesture and action recognition in depth image sequences. We introduced a novel framework, in which sequence of local oriented displacement features are modeled temporally as subspaces lying in Grassmannian manifold. We then formulated our learning algorithm using the notion of the class-specific tangent space on the Grassmannian manifold. Thanks to statistical tools applied on this riemaniann manifold, separate tangent space is considered

for each class and the classification process is performed as a function of probability density by Truncated Warped Gaussian on specific-class tangent spaces. The evaluation of our approach en terms of human activity recognition even in presence of object interaction and hand gesture recognition reveals a remarkable efficiency exceeding 95% on UT-kinect and MSR-Gesture3D datasets.

ACKNOWLEDGEMENTS

The authors would like to thank Anuj Srivastava for his assistance and useful discussions about this work.

REFERENCES

[1] S. Fothergill, H. Mentis, P. Kohli, and S. Nowozin, "Instructing people for training gestural interactive systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '12. New York, NY, USA: ACM, 2012, pp. 1737–1746.

[2] W. Lao, J. Han, and P. de With, "Automatic video-based human motion analyzer for consumer surveillance system," in *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, 2009, pp. 591–598.

[3] A. Jalal, M. Uddin, and T. S. Kim, "Depth video-based human activity recognition system using translation and scaling invariant features for life logging at smart home," in *IEEE Transactions on Consumer Electronics*, vol. 58, no. 3, 2012, pp. 863–871.

[4] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, 2010, pp. 9–14.

[5] A. Vieira, E. Nascimento, G. Oliveira, Z. Liu, and M. Campos, "Stop: Space-time occupancy patterns for 3d action recognition from depth map sequences," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, vol. 7441, 2012, pp. 252–259.

[6] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu, "Robust 3d action recognition with random occupancy patterns," in *Computer Vision ECCV 2012*, 2012, pp. 872–885.

[7] X. Yang, C. Zhang, and Y. Tian, "Recognizing actions using depth motion maps-based histograms of oriented gradients," in *Proceedings of the 20th ACM international conference on Multimedia*, New York, NY, USA, 2012, pp. 1057–1060.

[8] O. Oreifej and Z. Liu, "Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '13, Washington, DC, USA, 2013, pp. 716–723.

[9] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *Machine Learning for Computer Vision*, vol. 411, 2013, pp. 119–135.

[10] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *Computer Vision and Pattern Recognition Workshops CVPRW*, ser. CVPRW '12, 2012, pp. 20–27.

[11] X. Yang and Y. Tian, "Eigenjoints based action recognition using naive bayes nearest neighbor," in *Computer Vision and Pattern Recognition Workshops*, ser. CVPRW '12, 2012, pp. 14–19.

[12] S. Azary and A. Savakis, "A spatiotemporal descriptor based on radial distances and 3d joint tracking for action classification," in *19th IEEE International Conference on Image Processing*, ser. ICIP'12, 2012, pp. 769–772.

[13] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 1290–1297.

[14] P. Turaga and R. Chellappa, "Locally time-invariant models of human activities using trajectories on the grassmannian," in *IEEE Conference on Computer Vision and Pattern Recognition CVPR*, 2009, pp. 2435–2441.

[15] Y. M. Lui and J. R. Beveridge, "Tangent bundle for human action recognition," in *IEEE Int. Conf. Automat. Face Gesture Recog. FG*, 2011, pp. 97–102.

[16] S. Tang, X. Wang, X. Lv, T. Han, J. Keller, Z. He, M. Skubic, and S. Lao, "Histogram of oriented normal vectors for object recognition with a depth sensor," in *Computer Vision ACCV 2012*, ser. Lecture Notes in Computer Science, vol. 7725, 2013, pp. 525–538.

[17] P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa, "Statistical computations on grassmann and stiefel manifolds for image and video-based recognition," vol. 33, no. 11, Washington, DC, USA, 2011, pp. 2273–2286.

[18] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," vol. 20, no. 2, 1998, pp. 303–353.

[19] H. Drira, B. Ben Amor, A. Srivastava, M. Daoudi, and R. Slama, "3d face recognition under expressions, occlusions, and pose variations," in *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, 2013, pp. 2270–2283.

[20] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints." in *CVPR Workshops*, 2012, pp. 20–27.

[21] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo, "Space-time pose representation for 3d human action recognition," in *Workshop on Social Behaviour Analysis ICIAP*, Naples, France, 2013, p. 1.

[22] A. Kurakin, Z. Zhang, and Z. Liu, "A real time system for dynamic hand gesture recognition with a depth sensor," in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, 2012, pp. 1975–1979.

[23] A. Klaser, M. Marszalek, and C. Schmid, "A Spatio-Temporal Descriptor Based on 3D-Gradients," in *BMVC 2008 - 19th British Machine Vision Conference*, 2008, pp. 275:1–10.